IN THE CLAIMS:

Please amend the claims as follows:

1. (original) An interrupt delivery system, comprising:

a first pair of scaleable node controllers, wherein each of said scaleable node controllers supports at least one microprocessor;

a first scalability port switch coupled to each of said scaleable node controllers, wherein said first scalability port switch is to receive an interrupt request, determine an address of one of said scaleable node controllers from said interrupt request and transmit said interrupt request to said one of said scaleable node controllers.

2. (original) An interrupt delivery system as recited in claim 1, further comprising a peripheral component interconnect device.

3. (original) An interrupt delivery system as recited in claim 2, further comprising a peripheral component interconnect bus coupled between the peripheral component interconnect device and the first scalability port switch, wherein said peripheral component interconnect bus is able to support a plurality of additional peripheral component interconnect devices.

4. (original) An interrupt delivery system as recited in claim 3, further comprising a first input/output hub coupled between the peripheral component interconnect bus and the first scalability port switch, wherein said first input/output hub is able to support a plurality of additional peripheral component interconnect hubs.

5. (original) An interrupt delivery system as recited in claim 4, further comprising a second pair of scaleable node controllers, wherein said second pair of scaleable node controllers are coupled to said first scalability port switch.

6. (original) An interrupt delivery system as recited in claim 5, wherein the first pair of scaleable node controllers and the second pair of scaleable node controllers are coupled to a second scalability port switch.

7. (currently amended) An interrupt delivery system as recited in claim ~~5~~ 6, wherein the second scalability port switch is coupled to the first input/output hub.

8. (original) An interrupt delivery system as recited in claim 7, wherein the second scalability port switch is coupled to a second input/output hub, wherein said second input/output hub is able to support a plurality of additional peripheral component interconnect hubs and wherein each of the scaleable node controllers is coupled to four microprocessors.

9. (currently amended) A method for delivering an interrupt request in a multi-node computer system, comprising:

receiving an interrupt request at a scalability port switch;

determining a scaleable node controller to receive said interrupt request; and

transmitting said interrupt request to said scaleable node controller.

10. (original) A method for delivering an interrupt request in a multi-node computer system as recited in claim 9, further comprising determining a processor to receive the interrupt request.

11. (currently amended) A method for delivering an interrupt request in a multi-node computer system as recited in claim 10, further comprising comparing a priority of the ~~IRQ~~ interrupt request with a priority of the processor.

12. (original) A method for delivering an interrupt request in a multi-node computer system as recited in claim 11, further comprising interrupting the processor.

13. (currently amended) A method for delivering an interrupt request in a multi-node computer system as recited in claim 11, wherein said scalable node controller redirects the interrupt request through the scalability port switch to a different processor.

14. (original) A method for delivering an interrupt request in a multi-node computer system as recited in claim 9, wherein said interrupt request is a broadcast interrupt request.

15. (original) A method for delivering an interrupt request in a multi-node computer system as recited in claim 12, further comprising transmitting an end of interrupt to a correct interrupt controller.

16. (original) A method for delivering an interrupt request in a multi-node computer system as recited in claim 11, wherein the interrupt request is generated by a PCI device.

17. (original) A method for delivering an interrupt request in a multi-node computer system as recited in claim 11, wherein the interrupt request is generated by a processor.

18. (currently amended) A set of instructions residing in a storage medium, said set of instructions capable of being executed by a processor for searching data stored in a mass storage device comprising:

receiving an interrupt request at a scalability port switch;

determining a scaleable node controller to receive said interrupt request; and

transmitting said interrupt request to said scaleable node controller.

19. (original) A method for delivering an interrupt request in a multi-node computer system as recited in claim 18, further comprising determining a processor to receive the interrupt request.

20. (currently amended) A method for delivering an interrupt request in a multi-node computer system as recited in claim 19, further comprising comparing a priority of the ~~IRQ~~ interrupt request with a priority of the processor.

21. (original) A method for delivering an interrupt request in a multi-node computer system as recited in claim 20, further comprising interrupting the processor.

22. (currently amended) A method for delivering an interrupt request in a multi-node computer system as recited in claim 20, wherein said scalable node controller redirects the interrupt request through the scalability port switch to a different processor.

23. (original) A method for delivering an interrupt request in a multi-node computer system as recited in claim 18, wherein said interrupt request is a broadcast interrupt request.

24. (original) A method for delivering an interrupt request in a multi-node computer system as recited in claim 21, further comprising transmitting an end of interrupt to a correct interrupt controller.

25. (original) A method for delivering an interrupt request in a multi-node computer system as recited in claim 20, wherein the interrupt request is generated by a PCI device.

26. (original) A method for delivering an interrupt request in a multi-node computer system as recited in claim 20, wherein the interrupt request is generated by a processor.